Organizational Hierarchies for Real-Time, 4D Visualizations

Jacques Haus, Gregory Newton, William Ribarsky, Nick Faust, Larry Hodges
Graphics, Visualization, and Usability Center, Georgia Tech
Gregory Turner
Information Processing Branch, Army Research Laboratory

Abstract

This paper describes a 4D object hierarchy that is appropriate for the immense amount of time-dependent, 3D data associated with large scale simulations or information spaces. An organizational hierarchy approach is developed that can also handle elements not contained in the organization. Using this flexible approach, even complex scenarios with constantly moving, changing objects can be displayed and explored in real-time with minimal clutter. The user may also move objects around by direct manipulation, define paths, create or delete hierarchical elements, and make other interactions. To strengthen the capacity for distributed simulations and for using sensor information from multiple sources, DIS capability has been integrated with this approach for dynamic updates of position, direction and speed, and hierarchical structure. The paper will discuss how the techniques used here can be applied to widely different organizational structures.

Background and Need

If we can make use of all 4 dimensions available to us for our visualizations, and then navigate them in real-time, we have the potential to explore and understand the structure and content of even vast information spaces. This idea is applicable for datasets that have strong 3D spatial structure (so-called "landscape" data) but also for datasets that are large and multivariate, even without spatial structure. This latter are sometimes called "Information Landscapes". [1] The payoff here is in applying visual processing capability and high interactivity to exploratory visualization and analysis.

Although there has been work on 3D visualization of information spaces, including the use of organizational structures to establish hierarchies, less attention has been paid to how these structures can be employed to retain real-time behavior and reduce clutter, even for large collections of 3D objects. Robertson *et. al.* [2] made an important step in showing the effectiveness of 3D visualization with their work on *cone trees*. The cone tree is a 3D tree that moves and transforms to bring into focus nodes (with the surrounding tree region) that a user touches. Using the high interactivity and 3D structure that the method affords, cone trees have been applied to the visualization of large organizations, numbering 20,000 or more nodes. Recently, the Narcissus system [3] has been used to organize collections of objects based on semantic relationships and a set of repulsive and attractive forces. The interplay of forces result in a more visually organized structure, even when large numbers of objects are involved. Some other approaches also use the File System Navigator from SGI [4].

There is also need for 4D hierarchical visualization techniques in "constant information flow" situations, where the amount of information received can be large and varied. This is the case in modern military systems, among other applications, where sensing, networking, and information processing have been brought together. Information storage, computing, and networking technologies will be combined to form a "common picture of the battlefield" [5]--a picture shared by all ranks from infantrymen to overall commanders. The common picture of the battlefield will produce immense amounts of data associated with tactical goals and options, dynamic operations, unit and troop movement, and general battlefield information. A widely held objective is to display, update, and interact with these data in real-time using 3D visualization techniques and accurate terrain models. In this way military personnel can get quick and accurate overviews of the actual combat situation. "Real-time" here implies frame update rates of at least 15 frames per second. This puts a stringent time constraint on the system; all computation, rendering and display for each image must be completed in 1/15 second. All this detail must not overwhelm either the visualization hardware or the user, who must navigate the data in real-time and receive updates instantaneously.

In this paper we describe organizational structures for managing and displaying 4D visualizations in real-time. We have developed an approach that will be applicable to many types of organizational hierarchies. Using this approach, even complex scenarios with constantly moving, changing objects can be displayed and explored in real-time with minimal clutter. We discuss the hierarchies and how objects at various levels can be displayed based on field of view, distance to the viewer's eyepoint, and other factors. These factors are controlled based on organizational complexity to produce real-time visualizations. Selective queries may also reveal levels of the hierarchy regardless of the visual factors. The user may also move objects (at any level of the hierarchy) around by direct manipulation, define paths, create or delete hierarchical elements, and make other interactions. We show how information about hierarchical content, distribution, and movement can be revealed at all levels of detail. To make the approach concrete we demonstrate it with an example from battlefield information visualization. However we also indicate the flexibility of the approach by showing how it can be applied to wide area network visualization (possibly geographically based and with a constant stream of information about network traffic).

A main way this work differs from the information navigation and visualization methods mentioned above is in its consideration of detail management techniques and organizational methods for real-time visualization. It also addresses the question of handling dynamically changing hierarchical structures and objects within them that may move, appear, or disappear. Finally networking capability has been integrated permitting use of real-time data sent from a variety of sources including sensors, simulations, or other network nodes. This capability also will support a collaborative environment.

Representing Hierarchical and Other Information: Battlefield Visualization

Objects in a complex scenario may include those with a strong organizational hierarchy and those with much weaker organization. A strong hierarchy might be like the one in Fig. 1, with a definite number of levels and well-defined entities in each level. The military structure in Fig. 1 has a hierarchy consisting of unit -> platoon -> company ->, etc.... The units can be foot soldiers, jeeps, tanks, and so on. In representing military units, there are two possibilities: (a) you have access to all information concerning the unit and know to which branch of the hierarchy it belongs, which is often the case for friendly units (at least for your units); (b) you have incomplete or totally missing information concerning the branch of the hierarchy to which the unit belongs (example: enemy units).

Thus the battlefield scenario is a mix of hierarchical and non-hierarchical objects. This is typical of other types of scenarios as well. We can further distinguish battlefield information by dividing it into two types. (Other types could also be added.)

- Military Units (such as those described in the last paragraph)
- Control Measures

Control measures are different from the strongly organizational military units. They can be anything describing the battlefield setting or tactical information such as bridges, roads, mine fields, meeting points, air corridors, battlefield zone boundaries, etc. The "display" hierarchy here will be due entirely to spatial resolution. If the projected area of a control measure falls below a certain pixel threshold, the control measure will in most cases disappear rather than be subsumed into a group.

The units and control measure data are different from most geography-based data because the information is not permanently bound to a geographical location and can move around. This is especially so for the units and their hierarchy. In addition units and control measures may appear or disappear and should be movable by the user. Direct manipulation capability within the 3D environment is required for the latter.

Time Constraints. To maintain smooth navigation and immersion, a minimum rate of about 15 frames per second is the target. After deducting the time necessary for terrain retrieval, rendering, and other functions of the system, perhaps on the order of 10-30 milliseconds remain for symbology display. The point here is that, because of other demands on the time budget, only a fraction of each frame time remains for the symbology. We attack this problem by using our organizational hierarchy. At the leaf node level, when we are dealing with individual units, we may also use a display hierarchy coupled with multiple levels of resolution. We will certainly use this display hierarchy when we don't have an organizational structure. One question, which we will attempt to answer below, is whether the organizational hierarchy is rich enough to handle hundreds or more units (at the leaf node level) while still meeting our time constraints.

Organizational Hierarchy Implementation

VGIS System. To provide the context for battlefield visualization and as a vehicle to work towards a "common picture of the battlefield", we have built VGIS (Virtual Geographic Information System), within which we display the military object hierarchy. We call this "4D symbology" in analogy with the Army 2D symbology. VGIS is a large, multifaceted project to allow navigation of and interaction with very large and high resolution, dynamically changing databases while retaining real-time display and interaction. [6] The system allows users to navigate accurate geographies (less than 1 meter resolution in some cases) with sustained frame rates of 15-20 frames per second. The user can not only see these terrains from any viewing angle but also buildings, roads, high resolution imagery draped on the terrain, and other features. Underlying the system is a queryable GIS database [7] that can be accessed through direct selection in the 3D visualization. The paging, caching, and advanced detail management algorithms allow flythroughs of datasets of any size and extent. [8] In fact the system has been demonstrated using terrain and image data of over 20 GB. There are both VR and workstation-based versions of VGIS and an OpenGL version that will run on a variety of platforms. Among several new capabilities planned for the system is the capability to automatically generate 3D representations of particular urban areas from 2D road and feature databases. This will allow use for urban planning, evaluations of vegetation, soil, waterway, or road patterns, flood planning, and many other tasks.

Representation of Units. We have a rich set of representations at our disposal. The Army's 2D symbols [9] provide detailed symbolic representations for all units and all levels of the combat organization. We associate the 2D symbols with the 4D symbols so they can impart their information to trained observers. It is necessary to maintain contact with the training and knowledge that has gone into the 2D symbology. We have done this by attaching "signposts bearing 2D symbols to the 4D symbols. These signs, which can be turned on and off, are simple, flat surfaces that always face the viewer. Often nodes in a hierarchy have rich semantic content, some of which can be conveyed by signs like these (thus they are generally applicable to hierarchical visualizations).

Although some previous work has been done on the 3D display of hierarchies [2, 10], little has been done on applying hierarchical structures to objects that may require real-time updates such as military units and control measures. Research relevant to this area includes work on providing situation awareness in complex virtual environments [11]. This includes developing methods to take raw, unanalyzed data and rate its importance. These importance rankings are then used to rate activity around a series of "sentinels" placed by the user in a large scale battlefield scenario. The user then does not need to closely watch all the sentinel arrays simultaneously, but is rather alerted of important activity in any of them. In response the user can select the sentinel area and see the activity in detail. This approach, especially the methods for ranking activity, has relevance for our work.

Our representations follow a few rules:

(a) The use of color is reserved to denote the affiliation of a unit (friendly, enemy, unknown). No other information can clearly be associated with colors. This is a rule established by the Army.

(b) Objects representing higher levels in the hierarchy are graphically simple (cube, tetrahedron, etc.). One type is associated with each level in the military hierarchy, and size increases as one moves higher in the hierarchy. This rule serves two purposes: faster to render, and easier/faster to recognize which level is displayed. For the battlefield example, pyramids represent platoons, cylinders with octahedral cross-sections represent companies, cubes depict battalions, and larger cubes depict brigades. The use of different shape objects is quite effective. We find these shapes can be discerned even for quite small (far away) objects on the terrain. Fig. 2 shows the shapes used in the levels of the hierarchy.

There is more information available on each unit than just its graphical representation and its location. This information can be obtained by actively querying any unit. Some data can also be passively displayed by each unit through signposts hanging above it and depicting symbolic information.

One can also display the spatial and temporal distribution of children for a given level of the hierarchy. Without the cost of representing each subunit, a simple "footprint" gives their spread. Since units can be spread widely (over several square miles for mobile units in a battalion), this knowledge can be critical in determining the layout of the battle. Temporal information can be used to update footprint patterns so that they show the changing formations as the battle progresses.

We studied a number of representations for the footprints. For example, we looked at the idea of painting the terrain over which units were spread with a distinctive color or pattern. We also considered embedding a thin, translucent polygonal solid in the terrain. The translucent colored part sticking up out of the terrain would form a "3D footprint". The painting or texturing requires an involved texture mapping procedure that can be quite time-consuming, especially for several footprints. Also, both the painting and 3D footprint option could cover wide areas, perhaps obscuring the underlying terrain and features. We thus decided to use a simple set of 3D links or "pipes" to show the connections between a parent node and its children. We require the pipes to follow the terrain contours (so that they do not pass through hills, for example). Since it can be too time-consuming to query elevation for each node of high resolution terrains (which may be distributed at distances of 1M or less over hundreds of square miles), we use coarse resolution elevations from higher levels of the terrain quadnode structure. For this purpose we have modified VGIS with special fast calls to retrieve the coarse elevations. The pipes are placed far enough above the terrain that they do not embed themselves in higher resolution features. Fig. 3 shows a typical footprint structure (in this case between a motorized platoon, symbolized by a pyramid, and its vehicular units). An auxiliary question is where to place the parent node with respect to the rest of the footprint. If a command unit is identified, we place the parent over or near that unit. Otherwise, we use the centroid in the x-y plane to place the parent.

Level of Detail Management. As we have said above, our detail management includes both display (objects at multiple levels of resolution) and hierarchical schemes. We use both concurrently. Our choice of which scheme to employ is based on several factors including the user position and navigation speed (translation, rotation), the target frame rate, the object position on the screen, or a "benefit" weighting for the graphical object (representing its graphical complexity or importance). Since we integrate the symbology display into the VGIS system, we also use the viewing volume to determine what is visible and what is not for a given eyepoint. Only symbology within the viewing volume is drawn.

The level of the object detail or the organizational hierarchy that is displayed is constantly changing. From frame to frame, children can condense into parent symbols or parents can explode into their children. Thus at every frame there could be objects that pop-up or disappear. Our tests show that this causes two perceptual problems:

(a) Since a set of children and their parent are located at different positions, the relationship between the appearing and disappearing units is not always clear.

(b) Unit pop-up creates an "event" on the screen, which often unnecessarily attracts the attention of the user.

There are some possible solutions:

(a) Several levels of the hierarchy can be displayed together, including some clearly visible linking.

(b) Children can appear before the parent disappears, at the location of the parent, and then smoothly move to their real locations before the parent object disappears.

(c) Children can appear in some smooth transition from fully transparent to solid (as the observer navigates closer) at the ends of their footprint links.

We have done some empirical studies to show that these methods can be effective. These studies have been especially for options (a) and (c). However, more tests need to be done. Also a detailed procedure for applying these options while maintaining real-time operation needs to be established.

Features and Interactions. The symbology code is integrated as additional libraries in the VGIS system. It uses the VGIS user interface [6]: navigation (workstation window or immersive environment) and (noun-verb) interaction. The symbology libraries have a complete set of menu actions, which can be grouped in 3 sections.

- Information query. All represented units are interfaces giving access to entries in the database. This information can be accessed by selecting the unit. Currently the database is not heavily populated, but the query method is available and will become more important once large databases are connected.
- Display options. Since user needs vary, there are options to adapt the choice of displayed objects--like the number of levels of the hierarchy displayed together, the highest/lowest level of the hierarchy that can be displayed, the possibility of freezing the display, switches to show the 2D flags described above, the subunits footprints, and so on. One option allows display of the whole hierarchy at once, using the vertical dimension to show the hierarchy levels (only basic units are then positioned on the ground). It creates a very effective hierarchical tree, combining geographical information and hierarchical information. (See Fig. 2). Of course, this display method slows rendering since everything is visible, and is not suited for fast navigation.
- Interaction with units. All basic operation (add, move, remove, change affiliation, etc.) on units are possible.

Gaining the 4th Dimension—Including DIS

The Distributed Interactive Simulations (DIS) standard [12,13] has been established to provide a protocol that enables distributed simulations to communicate with each other. It has the potential to handle even large numbers of entities (>100,000) in a distributed environment. DIS reduces the amount of data transmitted through the use of local databases, best effort broadcasting, and dead-reckoning for simulated entities. Using local databases eliminates the need to transmit detailed descriptions of entity models and terrain data. Best effort broadcasting does not provide transmission verification, and therefore reduces the amount of network traffic by eliminating re-transmissions. Dead-reckoning further reduces network traffic by allowing a simulation to lower the frequency of object position updates. The simulation specifies which dead-reckoning algorithm should be used to estimate an object's position and only transmits position updates when the dead-reckoning model differs from the object's actual position by a threshold value. The use of the DIS protocol is not restricted to computer simulations. DIS is also intended to communicate information from real vehicles moving on instrumented ranges (e.g., the Army's National Training Center). Though developed for military applications, DIS has wide capability for other areas as well. Makers of 3D multi-user games, for example, are investigating its use.

DIS capability has been provided to the symbology libraries through the addition of two independent processes. The first process is the Loral DIS Network Manager. The Network Manager is responsible for receiving and buffering the multicast DIS packets from the local network. The second process is the DIS Receiver, which establishes a TCP/IP connection with the Network Manager and registers the type of DIS data that it is interested in. Since the Network Manager buffers the DIS packets, the Receiver is able to receive the data from the Network Manager asynchronously. The Receiver maintains information on all the entities in the simulation, and is responsible for performing the dead-reckoning calculations. The Receiver also performs the translation from the coordinate system used by DIS to the coordinate system used by VGIS.

The data provided by DIS is hierarchically organized and displayed within real-time constraints by using the level of detail methods described above. Time-dependent behavior extends throughout the hierarchy, since we are able to compute new positions for upper levels based on movements of the low-level units. Thus, for example, objects at all the levels displayed in Fig. 2 will move based on DIS updates. At the moment DIS does not provide organizational information. However, this is likely to change in future versions of the DIS protocol. Currently we provide a separate organizational structure for the DIS entities, but we will use information from the new DIS protocol and other sources in the future.

Some Scenarios

Battlefield Visualization. The scenarios described here use multiple resolution terrain elevation and imagery data from Fort Hunter Liggett in California. The full dataset is several kilometers on a side and has a maximum terrain resolution of 2 meters. Using our DIS capability and organizational management schemes, we have replicated a battlefield visualization with many units moving in a realistic fashion. The position of each unit is updated between every frame. These updates include horizontal translation and vertical positioning. For grounded units, vertical positioning is needed for both hierarchical position (based on level of detail) and terrain-elevation positioning. In this scenario two groups of units are depicted; friendly tanks and helicopters and enemy tanks. The friendly tanks are in a V-shaped formation, as are the enemy tanks. The helicopters are in a line formation. The opposing forces move towards one another and eventually interpenetrate spatial areas. (See Fig. 4.) Our detail management and navigation tools allow the user to fly around and observe this unfolding scene in real-time.

We have also developed a larger scale hierarchical scenario to further test our detail management schemes. This is a brigade-level (and above) depiction. It has 5 hierarchical levels showing brigade, battalion, company, platoon, and basic unit formations. All these are deployed on the Hunter-Liggett terrain. The total number of basic units is approximately 250. (See Fig. 5 for an overall depiction of the hierarchy.) The time to display each frame of the total symbology (with no organizational detail management) is about 0.1 second.

We have made performance tests with various size hierarchies to further examine our detail management schemes. For example, a standard brigade hierarchy would have brigade, battalion, company, platoon, and basic unit formations. The total number of basic units would be about 250. Results for a range of hierarchy sizes are given in Table 1, where rendering times with and without the organizational detail management are compared. A range is given for results with detail management since the rendering time will depend on unit density on the terrain. The rendering time with detail management can be more than a factor of 30 less than that without (for larger numbers of objects), and the latter is outside our time budget of 10-30 milliseconds in most cases. Since we have used the same display parameters (e.g., to decide when to open up a level of the hierarchy) for all the results in Table 1, we should certainly be able to further lower the rendering times in the second column through optimization and adaptive control of the parameters. Thus it appears our detail management schemes are adequate for real-time display of a wide range of hierarchy sizes.

No. of Units	Rendering time w/o detail management	Rendering time with detail management
100	0.002/ 0.002	0.025
300	0.005/ 0.01	0.14
800	0.01 / 0.03	0.4
1100	0.02 / 0.04	0.55
1400	0.025/ 0.05	0.7
1800	0.03 / 0.06	0.9
2100	0.03 / 0.065	1.05

Table 1 Comparison of rendering times for a range of hierarchy sizes

Network Visualization. We have constructed a completely different scenario to show the potential of our 4D organizational hierarchy approach. Our objective was to build a 3D hierarchical structure that would scale to include the whole Internet while still allowing focus on components at any level of detail. To do this we display, as shown in Fig. 6, each network hierarchy level as a ring in its own plane. Fig. 6 shows the view from the Atlanta SURAnet backbone node (Georgia Tech's node) with the Georgia Tech network displayed below. As one navigates closer to the Georgia Tech network its nodes could open up to reveal local networks in detail. This structure provides quick response and smooth movement; one can also easily the level of the hierarchy she is viewing because of the ring distribution. The network visualization tool also has a network finder capability whereby one enters a network name (which may be embedded deeply in the hierarchy); the system then navigates smoothly to the network location, preserving context. Of course, the Internet hierarchy could also be attached to appropriate geographic locations on a map, in which case it would be even closer to the battlefield visualization scenario above.

Conclusions and Future Work

We have developed a 4D detail management scheme that allows the mixing of hierarchical and non-hierarchical objects. The hierarchies can significantly reduce image clutter caused by many moving objects, can provide grouping structures to aid understanding, and offer significant control over detail so that real-time frame rates and interactions can be retained. DIS capability greatly strengthens the time dependent aspects of the hierarchical organization. It also allows the distribution, sharing, and joint visualization of entity data. With DIS our symbology/VGIS system immediately becomes a distributed system where multiple instances of the visualization can be run sharing common data and all having access to immediate updates.

We have worked out some scenarios for our detail management scheme. One shows time-dependent symbology in a realistic setting and also shows the use of an organizational hierarchy involving hundreds of units. For the latter, we demonstrate that our detail management scheme greatly reduces the rendering time for the symbology and that an organization involving hundreds of units placed on a realistic, high resolution terrain can be navigated in real-time. The organizational hierarchy structure is also flexible and general. It can be applied to many other types of data including those without a strong spatial dependence. To show this we present a scenario involving real-time navigation of the Internet structure.

We will continue working on several open questions. Our investigations will be extended to organizational hierarchies involving hundreds to thousands of moving units. The hierarchies will move in time and will change their structures. These studies will allow us to give our detail management approach a full workout and decide conclusively about its flexibility for handling large scenarios. We also expect to study further applications. These will include applications that are not geographically based and may not have any inherent 3D spatial composition. However they will have a strong organizational hierarchy. We will also look further at the question of building hierarchies from unstructured data, expanding on our capability to obtain and integrate organizational and structural data from different sources. Finally there is the question of adaptive detail management [14], which, as we have seen from our study of hierarchies with different numbers of elements, will be necessary to handle organizations that change significantly in number of element or in spatial distribution.

References

1. Keith Andrews. Visualising Cyberspace: Information Visualisation in the Harmony Internet Browser. *Proceedings of IEEE Information Visualization '95*, pp. 97-104.
2. G. G. Robertson, J.D. Mackinlay, S.K. Card, S.P. Robertson, G.M. Olson, and J.S. Cone Trees: animated 3D visualizations of hierarchical information. *CHI '91*, p.189-94.
3. R.J. Hendley, N.S. Drew, A.M. Wood, and R. Beale. *Proceedings of IEEE Information Visualization '95*, pp. 90-96
4. Joel Tesler and Steve Strasnick. FSN: The 3D File System Navigator. Silicon Graphics, Inc., Mountain View, CA, 1992.
5. Army Research Lab "Federated Laboratory Opportunity Conference". November 15, 1994 (Newark, New Jersey).
6. David Koller, Peter Lindstrom, William Ribarsky, Larry Hodges, Nick Faust, and Gregory Turner. Virtual GIS: A Real-Time 3D Geographic Information System. *Proceedings Visualization '95*, pp. 94-100.
7. Hanan Samet. Applications of Spatial Data Structures; Computer Graphics, Image Processing, and GIS. (Addison-Wesley, Reading, Mass, 1990)
8. Peter Lindstrom, David Koller, William Ribarsky, Larry Hodges, Nick Faust, and Gregory Turner. Real-Time, Continuous Level of Detail Rendering of Height Fields. To be published, *Computer Graphics (SIGGRAPH '96.)*
9. "Operational Terms and Symbols, Field Manual No 101-5-1," Headquarters, Department of the Army, Washington, DC, 21 October 1985.
10. George G. Robertson, Stuart K. Card, and Jock D. Mackinlay. Information Visualization using 3D Interactive Animation. *Communications of the ACM, 36* (4), pp. 57-71 (April 1993).
11. M. Stytz, E. Block, and B. Soltz. Providing Situation Awareness Assistance to Users of Large-Scale, Dynamic, Complex Virtual Environments. *Presence, Vol. 2,* pp. 297-313 (1993).
12. The DIS Vision - A Map to the Future of Distributed Simulation. Institute for Simulation and Training. Prepared by DIS Steering Committee, Version 1, May 1994, IST-SP-94-01.
13. L.B. McDonald, C.P. Bouwens, R. Hofer, G. Wiehagen, K. Danisas, and J. Shiflett. Standard Protocol Data Units for Entity Information and Interaction in a Distributed Interactive Simulation. *Proceedings of the Twelfth Interservice/Industry Training Systems Conference*, Orlando FL, pp. 93-102.
14. Thomas A. Funkhouser and Carlo H. Sequin. Adaptative Display Algorithm for Interactive Frame Rates During Visualization of Complex Virtual Environments. *Computer Graphics Proceedings*, Annual conference Series, pp. 247-254 (1993).

Figure Captions

Figure 1 Partial view of a military hierarchical structure.

Figure 2 View of a hierarchy displayed in the geographical context given by VGIS.

Figure 3 Representation of vehicular units and their organization in platoons (denoted by pyramids).

Figure 4 View from a simulation scenario (updates provided using DIS).

Figure 5 Scene from a real-time scenario with over 2000 units at the lowest level.

Figure 6 Organizational hierarchy for an Internet navigator.